DOCUMENT RESUME

ED 396 666 IR 017 842

AUTHOR Taylor, Harriet G.
TITLE A Team Software Development Approach for NCATE Computer Science Teacher Preparation.
PUB DATE 94
NOTE 7p.; In: Recreating the Revolution. Proceedings of the Annual National Educational Computing Conference (15th, Boston, Massachusetts, June 13-15, 1994); see IR 017 841.
PUB TYPE Reports - Descriptive (141) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS Accrediting Agencies; *Computer Science Education; *Computer Software Development; Guidelines; Higher Education; *Skill Development; Student Projects; *Teacher Education; Teaching Methods; Teamwork
IDENTIFIERS *Collaborative Learning; *National Council for Accreditation of Teacher Educ

ABSTRACT

Secondary computer science teachers who attended a special institute took part in a coordinated team software development exercise. The exercise was intended to provide experiences with collaborative learning in computer science, particularly to provide project design experiences and to allow students to practice new methodologies that could be employed in their classrooms. This exercise also addresses many of the computer science competencies prescribed by the National Council for Accreditation of Teacher Education (NCATE). This paper provides an overview of the team project methodology and of its fulfillment of the NCATE guidelines. (Author/BEW)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED 396 666

Paper (M2-201A)

A Team Software Development Approach for NCATE Computer Science Teacher Preparation

Harriet G. Taylor
Department of Computer Science
Louisiana State University
Baton Rouge, LA 70803
(504) 388-1495
taylor@bit.csc.lsu.edu

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
☐ This document has been reproduced as received from the person or organization originating it.
☐ Minor changes have been made to improve reproduction quality.
• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
Donella Ingham
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Key words: accreditation, computer science, NCATE, teacher preparation, team projects

Abstract

Educators in a special institute for secondary computer science teachers participated in a coordinated team software development exercise. The exercise was designed to provide real experience in an active computing environment and to allow students to practice new methodologies that could be used in their own classrooms. This collaborative learning approach to computer science education addresses many of the competencies prescribed by the newly-adopted NCATE computer science education standards. In this paper, the team project methodology for computer science education is presented and its relationship to the NCATE guidelines is explored.

Introduction

Educators preparing to enter secondary computer science classrooms face many challenges. They must acquire technical proficiency in computer science as well as learn the teaching and support methodologies needed to conduct and manage secondary computer science classes and laboratories. Often, educators look to local colleges and universities for programs and training to prepare them for teaching secondary computer science.

Currently, quality and content of computer science teacher preparation programs vary significantly across the United States. New standards (NCATE, 1993) adopted by the National Council for Accreditation of Teacher Education (NCATE) in 1993 provide models for computer science teacher preparation programs that will ensure quality preparation of all participating in such programs. Teachers completing programs based on these standards will participate in a coordinated group of experiences designed to develop competency as computer science professionals and as educators.

A group of secondary computer science educators participated in a special teacher preparation institute at a major university. One final, long-term project was a team software development exercise that included practice in analysis of computer science structures and concepts and exposure to special teaching methodologies that could be carried into the classroom. The exercise and experiences associated with the project addressed in some way many of the competencies found in the NCATE guidelines. In this paper, the NCATE guidelines and their relevance to computer science education are

IR017842

BEST COPY AVAILABLE

discussed, the team software development project exercise is described, and the relationship between this team approach and the NCATE guidelines is examined.

NCATE Standards

In October, 1993, the National Council for Accreditation of Teacher Education (NCATE) formally adopted standards for computer science education (Taylor, Thomas & Knezek, 1993a). These standards are subject-area standards that provide guidelines for programs that train secondary computer science educators. This action by NCATE, the only national organization in the United States authorized to accredit professional teacher preparation programs (NCATE, 1992), signals that computer science is a true distinct secondary discipline. Furthermore, computer science is a valid speciality area within the field of education with its own unique set of competencies and training requirements.

The NCATE guidelines (ISTE, 1993) are competency-based guidelines which are broken down into two parts: computer science content area preparation and professional teaching preparation. Unlike most older NCATE subject-area guidelines (NCATE, 1993), there are no prescribed courses. Instead, a coordinated group of experiences that build on each other is desired. There is no one place in programs where a specific competency must be addressed. Rather, the goal is that most competencies will be fulfilled by multiple experiences which may span several courses perhaps in both the professional education unit and the computer science unit.

The NCATE guidelines are new and revolutionary for computer science education. Computer science is very visible in American secondary education, but not in a formal way. Not all states offer teacher certification in this field, and those requirements vary greatly from state to state. Likewise, not all states count computer science as a valid secondary subject area. Studies involving computer science often are included within the business or mathematics curriculum rather than carry the label computer science. This formal recognition by NCATE, an organization with enormous respect and credibility within the field of education, may well help standardize many other aspects of secondary computer science beyond teacher training. Standard curricula and certification criteria should soon follow.

These guidelines represent a major step towards the establishment of a discipline. They provide a framework for teacher training programs. Still lacking are implementation models that meet the standards and that can be replicated by others. Researchers must continue to define these programs and the types of activities that are needed. In this paper, one such activity, which touches on over 10 of the competencies, is described.

Team Project Activity

Goals of Project

Educators enrolled in a special summer institute for computer science teachers participated in a coordinated team project exercise (Taylor, in press). This team project exercise was intended to provide experience with collaborative learning in computer science. Moreover, it was designed to achieve a number of important goals:

1. Serve as a mechanism for working with teachers at greatly differing competency and expertise levels to teach computer science concepts;
2. Provide "real" experience in computer science through active involvement in development and design of large projects beyond the scope of students working independently;
3. Teach students advanced computer science data structure concepts at a high level, including differentiation between structures and uses with respect to a given situation;
4. Develop the teachers' abilities to communicate effectively about the discipline;
5. Demonstrate important teaching methodologies for secondary education;
6. Give students practice in modelling teaching methodologies; and
7. Provide for professional growth as computer scientists and educators.

The activity was an overwhelming success by all criteria. Most of the educators had significant prior experience in writing programs and using computer languages. Most had limited contact with other computer professionals. None had ever participated in any type of computer science collaborative activity. They all left the course identifying it solely with this exercise and talking about the ways that their own teaching styles and approaches might change as a result.

BEST COPY AVAILABLE

Description of Activity

Students were assigned to teams of three people to work on team software development projects. The products were to be menu-driven programs that met a set of specifications handed out by the instructor. Each project had an intended audience or set of users in mind. The interface that was provided for the user was critical to the success of the projects.

The instructor played an active role in team management. A schedule was created which provided for daily team meetings to plan and review progress. The instructor "client" was on hand to answer questions about design features that were under consideration. Particular attention was to be paid to the software engineering aspects of the project. Design of appropriate testing strategies was part of the exercise. The schedule provided for a required testing phase with input from other teams several days before completion.

The deliverables included not only the final program with user and programmer documentation, but also a written report. The report was to identify the higher-level aspects of the project beyond the description of the lines of code or modules in the program. Items to be discussed included:

1. Problem analysis and data structure selection and organization;
2. Description of how the team functioned and developed the product to best utilize the skills of the various team members;
3. Problem areas or critical design decisions;
4. Limitations of the version of the product that was delivered;
5. Possible extensions to the product and similar problems that might be used as classroom assignments; and
6. Types of testing done on the product, potential bugs, and fail-safe procedures that were installed.

The final component of the exercise was a one-hour oral team presentation of the project. These presentations were to be done from the perspective of presentations to co-workers about on-going projects. They were to discuss high-level details similar to those in the written report as well as demonstrate the functionality of the project. These presentations were given as formal presentations in the departmental conference room rather than the regular classroom. The conference room contained various types of audio-visual equipment and presentation aids, including a computer with a data projection device. Many of the teachers got their first experience using such technology in support of their teaching field through this exercise.

Change of class atmosphere

Most of the teachers were skeptical but enthusiastic about the activity. Most had assigned team projects to their classes but had never participated in them. This activity had much more structure and challenge than the normal methodology of assigning the project and the team members and collecting the finished project at a later time. There was constant and lively interaction between team members, between the various teams (each working on a different project), and the instructor client. Teams bonded as units and worked on this and other problems well past normal class time on most days. The class assumed an excited atmosphere not often seen when working with similar groups.

These teachers displayed an unusual zeal for this activity. They had enormous pride in what they had accomplished and expressed new vitality for using similar collaborative learning activities in their own classes. This aspect may be somewhat unique to computer science education. Most secondary computer science teachers are teachers who have retrained from some other primary teaching field. Often, the field is one such as mathematics, where individual work rather than collaborative learning is valued. An earlier study of successful computer science teachers (Taylor & Magoun, 1993) showed that many of these teacher had not used collaborative teaching techniques in their first fields and had only learned the value of such approaches in computer science education through experience. This finding was indeed verified with the group of teachers who participated in these team activities.

NCATE Team Project Requirement

The NCATE Computer Science Education Endorsement Guidelines (ISTE, 1993) subdivide specialty content preparation in computer science into five components. Component four calls for experience in team software development projects. The team activity described in this paper is one of many possible experiences that could be used to document this proficiency.

The team project requirement was one of the most controversial of all of the items in the NCATE guidelines (Taylor, Thomas & Knezek, 1993a; Taylor, Thomas & Knezek, 1993b). Reviewers were either wildly in favor of the requirement or adamantly against it. Some readers were confused as to the exact meaning of the requirement and what types of experiences

were envisioned. One extreme might be a senior-level computer science analysis and design course. Few educators have or will have sufficient computer science background to participate in such courses. Another possibility might be a team multimedia courseware development project in conjunction with an educational technology course. A wide range of experiences is possible. The exact number of such experiences and depth is left to be defined.

Computer scientists usually work in collaborative atmospheres. Few work independently without consulting others or interfacing with them. The ability to work within a group and manage group activity is therefore fundamental to a computing professional. Likewise, it is fundamental for secondary computer science teachers. They are often the first computing professionals that pre-college students encounter. Their students' ideas and images of the profession are frequently formed from these experiences. In addition, resources in most secondary schools are limited and students are plentiful. Group activity is often a necessity just to allow all students access to equipment and to manage classroom and lab time effectively.

Other NCATE Competencies Addressed

This activity could be cited as evidence of partial fulfillment of many other NCATE competencies in addition to the team software development project requirement. These include competencies in both the computer science content area preparation section and the professional teacher preparation section. The following two sections identify and discuss some of the competency guidelines that could be addressed by this exercise.

Computer Science Content Area Guidelines (ISTE, 1993; NCATE 1993)

Item 2.1.1 *Functional knowledge of programming in a high-level language, program design, and verification methodologies.*

- The project provided hands-on experience in programming, design, and verification. The project was a concluding exercise to a two-course sequence in which the language Pascal was used as a programming tool. In particular, this exercise is well-suited to address the question of the knowledge level required by the competency. The exercises requires demonstration of functional knowledge of all of the concepts outlined.

Item 2.1.2 *Advanced knowledge of data structures and algorithm analysis.*

- The projects that were assigned all required advanced knowledge of data structures and principles of data abstraction. They all involved applications best suited to binary tree structures and contained significant file organization and management components.

Item 2.2 *Programming and laboratory experience to demonstrate advanced knowledge in at least two high-level programming languages.*

- Participants in this exercise demonstrated advanced knowledge and proficiency in one high-level language. Experience in a second language would still be necessary.

Item 2.5 *Development of written and oral communication skills related to the field of computer science.*

- Students discussed computer science concepts and issues during team meetings. The project required a written report and oral presentation that addressed computer science issues related to the project. The activity might fulfill the part of the requirement calling for at least one oral presentation. It is good preliminary background to a course that includes the required research paper.

Professional Teaching Preparation Guidelines (ISTE, 1993; NCATE, 1993)

Item 3.8.1 *Identification and modeling of communication of concepts.*

- The team meetings and discussion component of this exercise in addition to the oral reports provided the students with far more expertise in this area than hours of lecture might have achieved.

Item 3.8.2 *Identification and modeling of a variety of teaching and grouping strategies.*

- Again, one specific strategy was explored. It is reasonable to expect that several activities would be needed to fully document this item in the NCATE guidelines.

Item 3.8.3 *Functional knowledge of strategies for dealing with different learning styles and diverse populations.*

- The students were challenged to work as teams. Each team was selected to include individuals with vastly different learning styles and abilities. Teams also included both male and female members. One of the main chores for the teams was to determine ways to utilize each individual to help the team. This often resulted in one of the team members actually teaching the others some concepts or programming

BEST COPY AVAILABLE

practices. Certainly this activity does not reflect the full scope of this particular guideline, but it could be cited as partial evidence of proficiency.

Item 3.8.4 *Functional knowledge of methods of assessment and appropriate feedback techniques.*

- Students were actively involved in providing feedback, both during team meetings, testing sessions, and during the conference-like oral presentations. As the exercise progressed, the teachers acquired increasing proficiency in expressing their concerns and problems and interacting in positive ways about the projects.

Item 3.9 *Functional knowledge of designing, developing and evaluating laboratory activities for the computer science classroom.*

- Students acquired knowledge about the design of lab activities and explored possible similar activities that might be used successfully in the secondary classroom. They also identified and discussed ways such activities could be used in conjunction with other subject areas to enhance learning. The central question in this competency is the level of knowledge. There was significant discussion of key issues involved in this item. The only actual exercise that required some demonstration of this proficiency was a component of the written report. Again, it is uncertain that this exercise alone would achieve the level of competency required to completely fulfill this item.

Item 3.10 *Development of laboratory management skills necessary to support computer science classroom activities.*

- The teachers managed their own machines and time. During the course of the exercise, a strange, almost lethal virus attacked the lab and some of the students' home computers. We were all involved in identifying the virus, eradicating it, and protecting the products from future contamination. Likewise, the students managed the equipment for the oral presentations and the sharing of files between multiple machines. Each team developed its own management plan and also learned aspects of working as facilitators of learning from the model presented by the instructor client in the exercise.

Conclusions

The recent NCATE-approved computer science education guidelines provide standards for secondary computer science teacher training programs. Programs must involve coordinated experiences in both computer science as a subject area and teacher preparation specifically directed at the secondary computer science classroom. Universities seeking NCATE accreditation must prepare folios (Abramson, 1993; Thomas, 1992) that document experiences within their computer science education specialty programs that provide the level of knowledge or competency required by each of the standards. Often, multiple experiences will be cited in fulfillment of one particular item.

Quality computer science education programs will include activities that integrate the subject area with the teaching methodologies associated with it. Computer science teachers often bring with them methodologies learned from other content areas and must be exposed to methodologies associated with the field of computer science as well. The team project activity described in this paper is one type of activity that achieves this goal. It combines content and methodology. It builds on prior experiences and learning. It could be used as partial evidence in fulfilling many of the NCATE competency areas.

Computer science education is still in its infancy. Little is known through research about its methodologies. The project described here involves a fairly standard method of the computer science classroom. Other researchers must now develop and test methodologies and disseminate this information as we document the true content of the field of computer science education.

References

Abramson, G. W. (1993). Technology and teacher education accreditation process. *Ed-Tech Review*. Autumn/Winter, 27-30.

ISTE Accreditation Committee. (1993). *Curriculum Guidelines for Accreditation of Education, Computing, and Technology Programs*. Eugene, OR: ISTE.

National Council for Accreditation of Teacher Education. (1993). *Approved Curriculum Guidelines*. Washington, DC: NCATE.

National Council for Accreditation of Teacher Education. (1992). *Standards, Procedures, and Policies for the Accreditation of Professional Education Units*. Washington, DC: NCATE.

Taylor, H. G. (In press). The use of team projects to teach computer science concepts and teaching methodologies. *Journal of Computers in Mathematics and Science Teaching*.

Taylor, H. G. & D. Magoun. (1993). Grouping and laboratory management strategies for pre-college informatics educators : Studies of successful teachers. ***Rethinking the Roles of Technology in Education—10th International Conference on Technology in Education***. Cambridge, MA, pp. 749-751.

Taylor, H. G., L. G. Thomas & D. G. Knezek. (1993a). NCATE adopts computer science education standards. *Journal of Computer Science Education*. 7(3), 25-28.

Taylor, H. G., L. G. Thomas & D. G. Knezek. (1993b). The development and validation of NCATE-approved standards for computer science teacher preparation programs. *Journal of Technology and Teacher Education*. 1(4), 319-333.

Thomas, L. G. (1992). NCATE accreditation: Administering the ISTE guidelines. *Educational IRM Quarterly*. 2(2), 9-14.